# PATENT APPLICATION OF

# DEBARAG N. BANERJEE

# FOR

# IMPROVING DATA THROUGHPUT OVER LOSSY COMMUNICATION LINKS

## CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims priority from U.S. Provisional Patent Application No. 60/266,719 filed February 5, 2001, which is incorporated herein by reference.

## FIELD OF THE INVENTION

The present invention relates generally to data communication networks such as local area networks and the internet. More specifically, it relates to techniques for improving data throughput over lossy communication links, such as wireless links, within a data communication network.

## BACKGROUND OF THE INVENTION

Transmission Control Protocol (TCP) is the most common transport protocol currently used in data communication networks such as the internet. TCP is designed for use in traditional wired networks in which the dominant cause for packet losses and delays is congestion. Congestion occurs, for example, when a transmitting host sends too many packets too quickly over the network to a receiving host, causing some packets to be lost because the receiving host cannot handle such high throughput. In TCP, the receiving host is expected to send acknowledgement signals back to the transmitting host in order to verify that packets have been properly received.

According to the TCP protocol, the receiver is allowed to send various types of acknowledgment. A cumulative acknowledgment is an acknowledgment of an entire sequence of data bytes, up to a specified octet sequence number. A cumulative acknowledgment indicates an octet sequence number of the next expected data byte. If a packet is received out-of-order (i.e., the first sequence number of a new packet does not match the sequence number of the next expected byte), then the receiver normally repeats the most recent acknowledgement. Such a duplicate acknowledgments indicates to the transmitter that a packet was lost. It also assumes the packet was lost if the transmitting host receives no acknowledgement before the expiration of a timeout interval.

In TCP, rather than waiting for an acknowledgment of each packet before sending another packet, the transmitting host continues to send packets, stopping only if the number of unacknowledged packets exceeds a specified window size. The transmitting host dynamically adjusts the window size using a flow control technique. When there is no loss, the window size is increased. When a loss occurs, the window size is reduced. The particular flow-control technique determines how exactly the window size is reduced upon detection of a congestion error and how the window size back is again increased. For example, TCP-Reno performs a slow-start followed by congestion avoidance in response to time-out, and performs fast-retransmit (i.e., congestion) in response to three duplicate acknowledgements. TCP-Tahoe performs a slow-start followed by congestion avoidance in both cases. The slow-start technique reduces the window size to one and then doubles the window size after each successful transmission.

Because TCP was developed primarily for use on wired networks, it does not perform well when the network includes lossy links, such as wireless links. The primary reason for its poor performance is because TCP flow-control techniques assume that all types of packet loss are due to network congestion. In the case of a wireless link, however, packets may also be lost because of the lossy nature of the connection, and not only because of congestion. Because TCP assumes that congestion is always responsible for losses, it inappropriately responds to packet loss due to wireless fading as if the loss were due to congestion, e.g., by reducing the window size. This rate back-off mechanism causes the TCP throughput to degrade unnecessarily. Consequently, TCP performance in networks having wireless or other lossy links suffers from throughput degradation due to the inappropriate use of congestion compensation techniques.

Known techniques to improve the performance of TCP over wireless links all require both modifications to the wireless host, and also modifications to the wired host and/or to the wireless gateway node. These prior approaches include, for example, transport layer schemes that require alterations to the TCP implementation. Unfortunately, such alterations can lead to network deadlock and might otherwise be impractical due to backward incompatibility with already established internet hosts. Other approaches are split-connection techniques and link-layer techniques. The split-connection approach breaks the TCP connection into two separate connections: a wired connection between the wired host and the wireless gateway, and a wireless connection between the wireless gateway and the wireless host. This approach suffers from the lack of end-to-end semantics since acknowledgements received at the wired host only indicates reception by the wireless gateway, and not necessarily reception by the wireless host. This approach also has the disadvantage that it requires modification of all wireless gateways to manage the split connections. The link-layer techniques use link-layer mechanisms to make the link layer packet loss probability comparable to that of the wireless link. This improves the transport layer interaction at the cost of link layer throughput, and can be inefficient under some circumstances. This approach also has the disadvantage that it requires modification of all wireless gateways to manage the split connections. All the above approaches have the disadvantage that they require modification to the TCP implementation in hosts of the existing wired network, and/or modifications to the wireless gateway.

## SUMMARY OF THE INVENTION

In contrast with the prior art, the present invention provides a method for improving throughput in a packet data network without modifying the wired hosts or the wireless gateways. The present invention provides a novel and advantageous approach that only requires modification to the wireless host, and provides full compatibility with existing wired networks and wireless gateways. Also, most prior art involving modifications to the wireless gateways do not effectively address the problem when a multitude of wireless gateways may be present in the path(s) of the packets being transferred between the two hosts.

In a data network comprising a first host (e.g., a wired host), a second host (e.g., a wireless host), and a data connection subject to transmission errors (e.g., including a wired network, a wireless network, and one or more wireless gateways spanning the two), the method is implemented at the second host. The host first determines whether error-induced losses or congestion-losses dominate the data connection. If congestion-losses dominate the data connection, then the host uses a standard technique for acknowledging data packets. If, on the other hand, error-induced losses dominate the connection, the host sends a plurality of non-duplicate acknowledgements of a single packet whenever a packet is received after an out-of-order packet is received. By acknowledging distinct fragments of the packet, rather than identical (i.e., duplicate) acknowledgments of the packet, these non-duplicate acknowledgments of the same packet have the effect of accelerating recovery of maximal window size. In the case of domination by error-induced losses, the host also preferably adjusts the receive window length according to a capacity of the data connection.

In a preferred embodiment of the invention, the host determines whether error-induced losses or congestion-losses dominate the data connection by calculating a temperament parameter characterizing the error-proneness of the data connection. The temperament parameter preferably comprises taking the product of a packet error rate and a square of the delay-bandwidth product.

## BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of an exemplary data communications network in which a preferred embodiment of the present invention may be realized.

FIG. 2 is a flow chart illustrating steps in a packet transfer protocol method according to an embodiment of the present invention.

FIG. 3 is a flow chart illustrating details of a step in FIG. 2 which handles a packet according to a modified TCP method, according to an embodiment of the present invention.

## DETAILED DESCRIPTION

Particular embodiments of the present invention will now be described in detail with reference to the drawing figures. FIG. 1 shows a data communications network in which the

present invention may be typically realized. A first host 10 is connected via a wired network 20 to a wireless gateway 30. Wireless gateway 30, in turn, is connected to a second host 40 via a wireless network 50. Wireless network 50 comprises of at least one lossy link subject to transmission errors caused by physical effects such as fading, interference, or multipath. In contrast, the link or links between the first host 10 and the wireless gateway 30 are wired links that are not normally subject to significant transmission errors. Host devices 10 and 40 thus transmit and receive digital information through a combination of wired and lossy link segments. The first host 10 is also referred to as a wired host, while the second host 40 is referred to as a wireless host. Wireless gateway 30 is not necessarily unique. There may be one or more other wireless gateways, such as wireless gateway 60, which may provide alternate paths for communication between hosts 10 and 40. It should be understood that the data network of FIG. 1 is a simple illustrative example and that the actual implementation may be more complex. It should also be understood that the wireless gateway and the wired host may be, in some cases, the same machine. In other embodiments, in place of wireless network 50 is a network that contains at least one lossy link, and the second host is not a wireless host but a host connected to the first host via the lossy link.

In the preferred embodiment of the invention, a data connection is established between hosts 10 and 40, and a novel method is used at the host 40 to control the transfer of data packets over the connection. At the wired host 10 standard TCP is used. At the wireless host 40, however, a modified TCP technique is implemented. The modified TCP technique is designed to increase net throughput above the lower levels that would exist if standard TCP were used. It is a central advantage of the present invention that this modification to the TCP need only take place at the wireless host, and is fully compatible with the standard TCP operating at the wired host. Thus, no modifications to the wired host (or gateway) are needed for the improvement in TCP performance. It should be understood, however, that it is not a requirement that standard TCP be used at the wired host.

A preferred embodiment of the method implemented at the wireless host is illustrated in the flow chart of FIG. 2. The method includes a step 100 wherein a temperament parameter $\theta$ is calculated. The temperament parameter is herein defined generally as a quantitative measure characterizing the error-proneness of the data connection. In the preferred embodiment, however, temperament is not a direct measure of the probability of packet failure due to

error-induced losses (i.e., due to errors caused by noise, interference, fading, etc. on the lossy link). Instead, temperament is a measure of the contribution of error-induced losses to the reduction of TCP throughput as compared to the contribution of congestion losses. Thus, as the temperament increases, error-induced packet losses become the dominant cause for initiation of window cutback. In a preferred embodiment, the temperament parameter $\theta$ is calculated by taking the product of a packet error rate $q$ and a square of the product of link bandwidth $\mu$ and round-trip delay $T$. In other words, $\theta=q(\mu T)^2$. More specific details regarding the calculation of $\theta$ are included later in this specification.

After the temperament parameter has been calculated, control shifts at decision step 110 depending on whether error-induced losses or congestion-losses dominate the data connection. This decision is made based on a current value of the temperament parameter. If the temperament parameter exceeds a predetermined threshold value (typically 1), then the reduction of TCP throughput is due more to error-induced losses than to congestion. In this case, control is passed to a step 120 which handles packets according to a modified TCP. Otherwise, control is passed to a step 130 which handles packets according to standard TCP. In either case, after handling the packet, control is passed to a decision step 140 that determines whether or not a new value of the temperament parameter should be calculated. If so, control is passed to blocks 150 and 100. Otherwise, these blocks are by-passed and control is passed directly to block 110. The temperament is preferably computed whenever a connection is initiated, and re-computed during the session to reflect changes in the lossy link. The temperament can be re-computed whenever there is a handoff or other major event, whenever a packet is sent, and/or at fixed time intervals. The particular network characteristics will determine the best conditions for deciding at block 140 whether or not to re-compute the temperament. For example, if the link has a guaranteed quality of service, then it may be sufficient to re-calculate the temperament only for handoff or other major events. When there is no guaranteed quality of service, it is preferably to have more frequent updates of temperament.

The modified TCP protocol of block 120 is identical to standard TCP except for the following details, which are illustrated by the flow chart in FIG. 3. Whenever the wireless host receives a new packet, it checks if the new packet follows an out-of-order packet (decision block 200). If not, control is transferred to block 210 and the packet is handled according to standard TCP. If the new packet does follow an out-of-order packet, however,

then control is transferred instead to block 220 where the wireless host sends multiple non-duplicate acknowledgements of the new packet to the wired host. The effect of this is to create a fast rate recovery to compensate for the back-off caused by the initial out-of-order packet. When the out-of-order packet is first received, the wireless host sends duplicate acknowledgements in accordance with the standard TCP fast-transmit algorithm. The wired host interprets these duplicate acknowledgments as an indication of congestion, resulting in a reduction in window size. However, when a new packet is received following the out-of-order packet, the wireless host sends a series of non-duplicate acknowledgments to this single packet, which the wired host interprets as acknowledgments of multiple received packets, resulting in a fast increase in the window size. This artificial quick-recovery happens only when the connection is dominated by error-induced packet losses and the modified TCP is being used. The conventional TCP rate-backoff and recovery algorithms are preserved by the present invention, the key difference being that the recovery is artificially accelerated in the case of domination by error-induced losses. If congestion losses dominate, the method operates as standard TCP.

This modified TCP technique exploits a feature of standard TCP in order to generate multiple non-duplicate acknowledgements of a single packet at the wireless host, thereby deceiving the wired host into quickly increasing the window size. In standard TCP, cumulative acknowledgments make reference not to packet sequence numbers, but rather to octet sequence numbers. Because each packet comprises a plurality of octet fragments, standard TCP permits multiple non-duplicate cumulative acknowledgments of a single packet. Such acknowledgments are also called fragmented acknowledgments, since they acknowledge distinct octet fragments within the same packet. For example, a packet starting with octet sequence number $S$ and finishing with octet sequence number $F$ may be acknowledged with $N$ fragmented acknowledgments making reference to octet sequence numbers $S+(F-S)/N$, $S+2(F-S)/N$, ..., $F$. By fragmenting the packet into octets, the $N$ acknowledgments of the same packet are non-duplicate, and are not interpreted by the wired host as indicating packet loss. Instead, they are interpreted as acknowledgments of separate packets, and result in accelerated increase in window size. Preferably, the $N$ fragmented acknowledgments are sent from the wireless host at equal intervals within a round-trip time $T$. This results in a dramatic increase in the recovery time as compared to standard TCP. The value of $N$ determines the rate of recovery that is desired, and is selected based on various factors that depend upon the specific nature of the data network, particularly the

characteristics of the lossy link. For example, in the case of TCP-Reno, in order to keep the TCP throughput degradation acceptable (say, within $\eta$ of the maximum throughput) the value of $N$ should be set in proportion to the temperament value $\theta$. In particular, $N=\theta/4\eta$.

Returning now to FIG. 2, at block 150 the host also preferably computes a receive window length according to a capacity of the data connection, and advertizes the length if it has changed. In conventional TCP, the wireless host normally would advertise its receive window to be equal to its available buffer space. In the case of a wireless link, however, the buffer space is not likely to be the bottleneck. Instead, the limited link capacity is likely to be the bottleneck. Thus, it makes better sense to set the wireless host receive window according to the capacity of the wireless link data connection rather than according to its buffer capacity, since this will result in less congestion. In other words, if the maximum data rate of the TCP connection over the wireless link is $R$, and the round-trip delay for the connection is $T$, then a window size larger than $W=RT$ will likely cause congestion. The round-trip time $T$ is known, and the maximum TCP connection rate $R$ can be calculated from the known values of the maximum link-layer rate $R_L$, frame size $FS$, maximum segment size $MSS$, and the IP packet overhead $IP$. In particular, $R=R_L/[(MSS+IP)/FS]$, where the square brackets indicate the ceiling of the quantity contained within. For multiple TCP connections sharing the same link, the window size $W_i$ for each connection $i$ can be scaled according to the connection's bandwidth-delay product. In particular, $W_i = RT_i ( T_i \mu_i / \Sigma_i T_i \mu_i )$, where $T_i$ and $\mu_i$ are the delay and bandwidth, respectively, for connection $i$. This adjustment of the receive window is preferably performed whenever the multiple non-duplicate acknowledgment technique is used in order to prevent the possibility of congestion breakdown of the IP network due to faster-than-expected recovery.

There may, in general, be many different ways to determine at decision block 110 whether error-induced losses or congestion losses are likely to dominate a lossy connection. In the preferred embodiment of the present invention, this determination is based upon a current value of a temperament parameter calculated at block 100. Other parameters or measures, however, are considered within the scope of the present invention. Preferably, any such parameter or parameters are measured or computed from information normally available at the wireless host, without requiring special requests for additional information from the wired host. For example, in the preferred embodiment of the present invention, the

calculation at block 100 of the TCP temperament parameter $\theta$ is based on the following information: The effective data rate over the wireless link, $\mu$, the round-trip time, $T$, and the random packet loss probability, $q$.

In standard TCP, the value of $T$ is known to the wireless host since it is an essential part of standard TCP. However, in an asymmetric link where the wireless host is primarily receiving, the estimate of $T$ may be inaccurate because a significant time may have passed since the wireless host last transmitted an original packet. To avoid this pitfall, the wireless host preferably sends a "ping" request, or similar benign message, whenever the last update to its round-trip estimate is past a predetermined threshold age. The wired host's acknowledgment to the message then provides an up-to-date estimate of $T$. The frequency at which these messages should be sent depends on the average rate of fluctuation in the network topology. If, for example, the wireless host performs a handoff, it should update the estimate of $T$ to reflect the new network topology.

The wireless host can easily estimate $q$ from frame error statistics at the data link layer and the known values of the segment size and frame size. The packet loss probability $q$ is defined as the probability of a maximum segment sized packet getting lost during transmission over the lossy link. It can be obtained by conjoining the frame error probability of the number of data link frames that would be needed to transfer such a packet. For example, let $M$ be the maximum segment sized packet in bytes (including TCP and IP headers), let $F$ be the data link frame size, having a frame error rate (after any FEC coding and/or ARQ) of $P_f$. Then $q=1-(1-P_f)^{[M/F]}$, where the square brackets again signify the ceiling of the quantity contained within.

The wireless host can determine $\mu$ from the packet arrival rate. Preferably, the value of $\mu$ is calculated at the wireless host transport layer by maintaining a running average of the number of octets of transport layer data (including acknowledgments and duplicate segments) arriving per second, which it will update every time a new packet arrives. This octet rate is then normalized with the maximum segment size to arrive at the desired packet data rate. Although this measure does not count the packets that get lost at the wireless link, this is not expected to cause any significant errors in the estimation since the packet loss probability $q$ is normally much less than 1.

Those skilled in the art will appreciate that many details discussed in relation to the above embodiments of the invention may be altered in various ways without departing from the essential features of the invention. For example, the principle ideas of the present invention will work on any transport layer mechanism where an end-to-end acknowledgment-based rate-backoff mechanism is used and where the acknowledgments are designed to be cumulative in nature and can acknowledge parts of packets. Thus, the above description represents a particular instantiation of the invention for TCP, and many other instantiations are possible. Accordingly, the scope of the invention is not limited to the specific details included above for illustrative purposes, but is determined from the following claims.